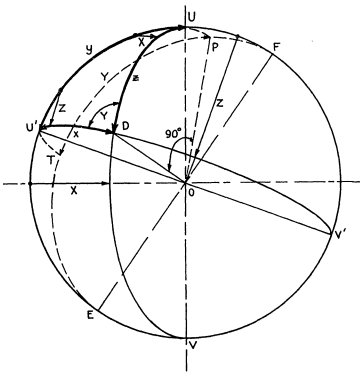
ROUND WORLDS

Lawrence Roberts, a pioneer of computer visualization at MIT, built an algorithm in the 1960's that is still used to develop 3-D visualizations in many software platforms. Importantly, Roberts work involved a translation of 19th century geometry into the matrices used to program computers. This act of translation rendered a new form visualization that allowed geometric objects to be displayed in a simulated 3-D space. Like Roberts, we have focused on translating old math into software. We translated the three-dimensional motion of spheres into a program that generates objects. However, unlike Roberts our programs remain usable only to the creator. They have yet to enter the "world". Each program contains idiosyncratic names and relationships. It is also based on an understanding of programming. It therefore remains inaccessible to many. It is also a conventional form of creative work, where the author produces a distinct object valued for its idiosyncratic qualities. Another model of creative work involves the crafting of a computation object that can be easily modified and transformed by a vast array of collaborators and users. This model shifts the focus of the creative work to the development of shareable tool that is accessible to a wide variety of users. In the next and final assignment we will focus on translating our programming work into forms that will allow other users to create new objects and interact with our work in unexpected ways.



Milton Sphelsteing, On Spherical Drawing and Computation, 1955

1

TRANSLATION 1

Follow the in-class tutorials to build a plug-in for Rhinoceros 6.0. The plug-in should have single command with a series of prompts that require the user to input variables. Importantly, this is a design problem. Please consider the amount and type of information that you ask of the user. Do you need to change some of the variables in the program so that they make more sense to the user? What limits do you need to build into the program to prevent the user from freezing their computer or otherwise compromising the program? What instructions will be provided in the prompts?

Each plug-in should accomplish the following:

1. Operate as a command that can be called from the command line in Rhino without opening the Python editor.
2. Produce a Spherical object with a minimum of 2 colors and composed entirely of solids.
3. Capture a view of the object at a resolution of 1600 x 1600 pixels.
4. Export an .obj file of the object with an associated .mtl file
5. Save a text file with all the user entered data.

Roberts found that there was no known way of representing perspective on the computer. He thus hacked it by studying the mathematical methods for perspective geometry from German textbooks from the 1800s and then 'translating' them into the matrix form used to program computers...his algorithm continues to run in most modern 3-D

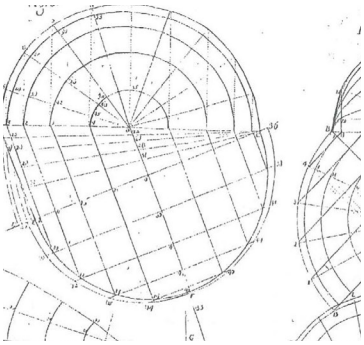
software.

(Daniel Cardoso-Llach, AD, 84, No2)

READING:

Peters, Brady,and Daniel Davis. "Design

Ecosystems:.83, no. 2 (April 2013).



Guarino Guarini, Architettura civile, 1735

After the completion of the plug-in please ask (3) students either from the class or outside of class to run the plug-in and return to you: (1) obj file, (1) png, and (1) txt file.

In addition, please produce (1) 2 minute tutorial of you operating the program and explaining how it works. Provide this tutorial to the user.

Please submit the plug-in , the collected work from 3 student, and your tutorial

Due: 4.05

TRANSLATION 2

Follow the in-class tutorials to translate the Rhino plug-in into a definition in Grasshopper for Rhino 6. Use the definition to build a gallery to exhibit your work from the semester in.

Due: 4.19

ROUND WORLDS EXHIBITION

Export the gallery as obj and open in blender. Use the Decimate modifier to reduce the polygon count and overall size. Export the gallery as a glb and load it into Mozilla Spoke as your gallery environment for Mozilla's on-line VR platform Hubs.

Import 4 models (minimum) from the semester into Blender, decimate the meshes as above, and add a keyframe animation of simple motion to each of the models. Try to keep the model size to less than 3 mb. Load the models into your Spoke Gallery and turn on the animations.

Import relevant links from (4) Animations and load (4) Drawings from the semester as assets in the Gallery. Publish the Gallery to hubs and send a link to me.

Place the glb files, animations, and drawings in a folder entitled "RoundWolds" inside of project 3 on the shared One-Drive.

Due 4.26